Study of Statistical Systems Using Monte Carlo Methods

Vibhu Gupta

February 25, 2010

It is of prime interest in various fields to study a system which constitute of very large number of particles. The physical properties of such a system cannot be deduced by solving the classical Newton Laws (or Schrodinger Equation for quantum systems) sheerly because of the complexity and the very large number of equations and even larger initial conditions. For this statistical methods become an indispensible tool .In statistical approach towards the problem, it is of immense interest to find the partition function from which other expectation values of observables can be easily calculated .Since partition function cannot be calculated analytically except for few special and approximated systems ,computational methods have to be used to investigate the properties of the system.

Ising model is one of the problems whose properties are studied extensively using computational techniques. Monte Carlo methods are extensively used as an effective way to study these system. The basic idea behind this method is to simulate the system to its equlibrium condition and find the corresponding transition rates between the different states of the system ,such that the distribution is boltzmann . The expectation values of the observables then can be eventually evaluated in a straightforward manner.

Metropolis algorithm is an effective algorithm to simulate ising model. The core idea behind this method is to randomly change the configuration of the system under the condition that either energy of the system decreases by the change ,or if the energy increases ,the change is taken with a probility defined by boltzmann distribution.

Random number generators are an essential part of this methodology. I will like to undertake the study of how these random number generators work and investigate for more efficient random number generators. I will like to understand the metropolis algorithm and investigate for its complexity and stability. I will like to look for possible parallelization of the algorithm ,its complexity and speed up.